

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-107465

(43)Date of publication of application : 21.04.1995

(51)Int.Cl.

H04N 7/24  
G06T 9/00

(21)Application number : 05-247881

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 04.10.1993

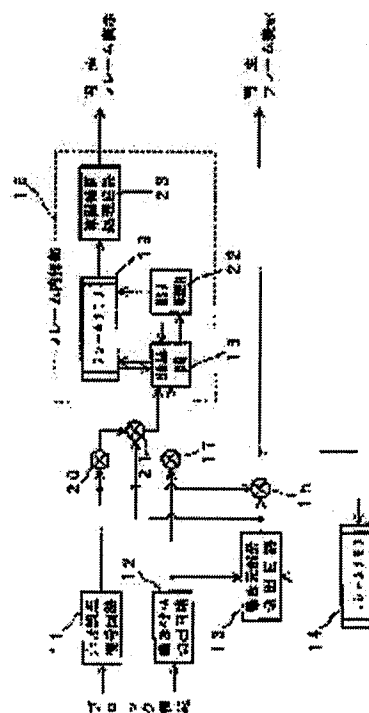
(72)Inventor : TSUKAHARA YURIKO  
KOBAYASHI HIROSHI

## (54) MOVING PICTURE FRAME REPRODUCING SYSTEM

### (57)Abstract:

**PURPOSE:** To synthesize an interpolated frame having the visually more natural movement by fitting a prescribed picture to the interpolated frame placed in the middle of the movement locus from the movement source to the picture part of the movement destination.

**CONSTITUTION:** A block error decoding circuit 11 decodes the block error from block information with respect to each block, and a motion vector decoding circuit 12 decodes the motion vector. A movement source part segmenting circuit 13 segments the movement source part from a frame memory 14 based on the motion vector, and an adder 15 adds the movement source part and the block error to reproduce the picture element value of the present block. A frame interpolating part 16 interpolates the picture element value of the interpolated frame by the present block. A fitting circuit 18 uses a vector, which is obtained by reducing the length of the decoded motion vector to 1/2 by a multiplier 17, to segment the interpolation part from a frame memory 19. The error obtained by reducing the decoded block error to 1/2 by a multiplier 20 and the picture element value of the movement source part are added by an adder 21, and an overlap processing circuit 22 processes the overlap part of picture elements.



## LEGAL STATUS

[Date of request for examination] 02.10.2000

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]	3441120
[Date of registration]	20.06.2003
[Number of appeal against examiner's decision of rejection]	
[Date of requesting appeal against examiner's decision of rejection]	
[Date of extinction of right]	20.06.2006

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-107465

(43) 公開日 平成7年(1995)4月21日

(51) Int.Cl.<sup>8</sup>

H 0 4 N 7/24

G 0 6 T 9/00

識別記号

庁内整理番号

F I

技術表示箇所

8420-5L

H 0 4 N 7/ 13

Z

G 0 6 F 15/ 66

3 3 0 H

審査請求 未請求 請求項の数 4 O L (全 13 頁)

(21) 出願番号 特願平5-247881

(22) 出願日 平成5年(1993)10月4日

(71) 出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72) 発明者 塚原 由利子

神奈川県川崎市幸区柳町70番地 株式会社

東芝柳町工場内

(72) 発明者 小林 浩

東京都港区芝浦1丁目1番1号 株式会社

東芝本社事務所内

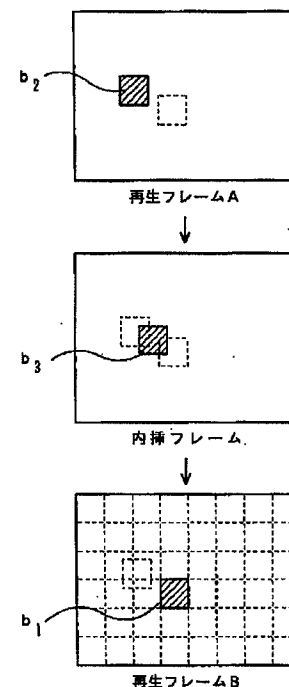
(74) 代理人 弁理士 須山 佐一

(54) 【発明の名称】 動画像フレーム再生方式

(57) 【要約】

【目的】 視覚的により自然な動きを持つ内挿フレームの合成を行い、符号化による生じるコマ落しを補間し、再生時の画質を向上させることができる動画像フレーム再生方式を提供することを目的としている。

【構成】 フレームB中のブロックb1の動き元部分がフレームA中のb2の部分であるとき、内挿フレーム中のb2からb1への動軌跡の中間に当たるb3の部分を、例えばb1とb2との画素値の平均値で内挿するものである。



【特許請求の範囲】

【請求項1】 コマ落としされた動画像のフレームを再生する動画像フレーム再生方式において、前記コマ落としされたフレームの前後のフレームに基づいて、前記コマ落としされたフレーム内の動き画像部分およびその動き軌跡を求める手段と、前記動き軌跡に基づいて、前記コマ落としされたフレーム内の動き画像部分のはめ込み位置を求める手段と、前記はめ込み位置に所定の画像を内挿または外挿する手段とを具備することを特徴とする動画像フレーム再生方式。

【請求項2】 請求項1記載の動画像フレーム再生方式において、前記はめ込み位置が隣接するはめ込み位置と重なった場合または隙間が生じた場合、前記はめ込み位置の周囲の画像と前記はめ込み位置の画像または前記はめ込み位置の周囲の画像を用いて補間処理をする手段をさらに具備することを特徴とする動画像フレーム再生方式。

【請求項3】 請求項1記載の動画像フレーム再生方式において、前記はめ込み位置が隣接するはめ込み位置と隙間が生じた場合、前記コマ落としされたフレームの前後のフレーム内の動き画像部分の周囲の画像を用いて補間処理をする手段をさらに具備することを特徴とする動画像フレーム再生方式。

【請求項4】 請求項1記載の動画像フレーム再生方式において、前記はめ込み位置に所定の画像を外挿することを、前記コマ落としされたフレームの前後のフレーム間でシーンチェンジがあった場合にのみ、または、前記コマ落としされたフレームの後のフレームのイントラ符号化された部分にのみ適用されるように制御する手段をさらに具備することを特徴とする動画像フレーム再生方式。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、動画像フレーム再生方式に関する。

【0002】

【従来の技術】動画像の符号化圧縮技術は、TV会議などの画像伝送やVTRなどの画像蓄積に不可欠なものである。すでに、国際標準の動画像符号化方式としてH.261方式と呼ばれる方式が勧告されている。

【0003】動画像のコマ数は、TV信号で約30フレーム/secであるが、伝送レートやメモリ量の制限から、符号化時にはコマ落としが起こるのが普通である。そこで、より自然な動画像を得るため、再生画像からフレーム内挿を行い、フレームレートを向上させる方式が提案されている。その一例として、「MC誤差を利用したフレーム内挿方式」（1990年電子情報通信学会秋季全国大会D-229、渡邊利明）を以下に説明する。

【0004】図12は同方式を実現するための概略的回路図である。なお、ここでは、簡単のため、フレーム内挿は2枚の再生フレームA、Bの中間の時点において1枚行うものとする。また、符号化方式としてはH.261方式を想定し、画像をブロック分割しブロック毎に動きベクトルおよび動き元部分と現ブロックとの誤差を符号化しているものとする。

【0005】同図に示す方式では、再生時にはブロック誤差復号回路1がブロック情報から順次ブロック誤差を復号し、動きベクトル復号回路2がブロック情報から順次動きベクトルを復号する。

【0006】次に、動き元部分切り出し回路3がフレームメモリ4に蓄えられているフレームAから動きベクトル復号回路2により復号された動きベクトルを用いて動き元部分を切り出す。

【0007】そして、加算器5において、この動き元部分にブロック誤差復号回路1により復号された誤差を加算してフレームBを再生する。

【0008】次に、乗算器6において、動きベクトル復号回路2により復号された動きベクトルを $1/2$ にし、動きベクトル補正回路7により隣接するブロックの動きベクトルなどを用いて補正を行う。

【0009】その後、動き元部分切り出し回路8がフレームメモリ4に蓄えられているフレームAから動き元部分を切り出し、加算器9および乗算器10においてフレームBの現ブロック部分との平均画素値をもって内挿フレームとする。

【0010】図13はこうしたフレーム内挿の原理を示す図であり、各フレームを一次元に投影して表したものである。

【0011】同図において、内挿しようとしているブロックをブロック3とし、後フレームBの同じ部分のブロックをブロック1とする。ここで、ブロック1の動きベクトルVによって前フレームAのブロック2の部分が切り出されるならば、ブロック3の画素値はベクトルV/ $2$ によって切り出されるブロック4の部分の画素値に近いと考えられる。よってこの部分をフレーム内挿に使用する。図14はこうした原理を二次元の絵で現したものである。

【0012】しかしながら、このような従来の動画像フレーム再生方式では、内挿フレームを互いに重ならない小部分（ブロック）に分割した後、各ブロックについて、後フレームの動きベクトルを用いて前フレームから似通っていると思われる部分を推定していたため、必ずしも推定が正しいとは限らず、ちぐはぐな画像を合成することが多かった。特に、動物物の境界に近い部分では、動きベクトルの変化が激しいため、推定が当たらないことが多かった。

【0013】

【発明が解決しようとする課題】このように、従来の動

画像フレーム再生方式は、内挿フレームを互いに重ならない小部分に分け、各部分について前フレームから似通っていると思われる部分を推定していたため、必ずしも推定が正しいとは限らず、ちぐはぐな画像を合成することが多かった。

【0014】本発明は、このような課題を解決するためになされたもので、視覚的により自然な動きを持つ内挿フレームの合成を行い、符号化による生じるコマ落としを補間し、再生時の画質を向上させることができる動画画像フレーム再生方式を提供することを目的としている。

【0015】

【課題を解決するための手段】かかる課題を解決するため、本発明の動画画像フレーム再生方式は、コマ落としされた動画画像のフレームを再生する動画画像フレーム再生方式において、前記コマ落としされたフレームの前後のフレームに基づいて、前記コマ落としされたフレーム内の動き画像部分およびその動き軌跡を求める手段と、前記動き軌跡に基づいて、前記コマ落としされたフレーム内の動き画像部分のはめ込み位置を求める手段と、前記はめ込み位置に所定の画像を内挿または外挿する手段とを具備する。上記の動画画像フレーム再生方式において、前記はめ込み位置が隣接するはめ込み位置と重なった場合または隙間が生じた場合、前記はめ込み位置の周囲の画像と前記はめ込み位置の画像または前記はめ込み位置の周囲の画像を用いて補間処理をする手段をさらに具備してもよい。

【0016】また、上記の動画画像フレーム再生方式において、前記はめ込み位置が隣接するはめ込み位置と隙間が生じた場合、前記コマ落としされたフレームの前後のフレーム内の動き画像部分の周囲の画像を用いて補間処理をする手段をさらに具備してもよい。

【0017】さらに、上記の動画画像フレーム再生方式において、前記はめ込み位置に所定の画像を外挿することを、前記コマ落としされたフレームの前後のフレーム間でシーンチェンジがあった場合にのみ、または、前記コマ落としされたフレームの後のフレームのイントラ符号化された部分にのみ適用されるように制御する手段をさらに具備してもよい。

【0018】

【作用】本発明では、内挿フレームを予め小部分に分割せず、動き元の画像部分から動き先の画像部分への動軌跡の途中に当たる内挿フレームの画像部分に所定の画像をはめ込むことにより、推定を行うことなく動きのスムーズな内挿画像を得ることができる。なお、このような内挿方式を用いると、画像部分をはめ込む際に重なりや隙間を生じるが、本発明では、例えば、重なりは平均を取ることで、隙間は近隣画素値を用いて補間することにより、自然さを損なわずに完全なフレーム内挿を行うことができる。

【0019】

【実施例】以下、本発明の実施例の詳細を図面に基づいて説明する。

【0020】図1は本発明の動画画像フレーム再生方式の原理を示す図である。

【0021】ここでは、簡単のため、フレーム内挿は、2枚の再生フレームA、Bの中間の時点について1枚行うものとする。

【0022】同図に示すように、本発明の原理は、フレームB中のブロックb1の動き元部分がフレームA中のb2の部分であるとき、内挿フレーム中のb2からb1への動軌跡の中間に当たるb3の部分、例えばb1とb2との画素値の平均値で内挿するものである。この内挿作業は、フレームBの全てのブロックについて行う。これにより、動きのスムーズな内挿画像を得ることができる。

【0023】このとき、動き元部分を指し示す動きベクトルは、各ブロックについて異なる方向を向いているので、内挿フレームを合成する際には、画素の重なりや隙間を生じる。そこで、本発明では、以下の処理を行う。

【0024】図2はあるブロックにより内挿された部分a2が、既に内挿されている部分a1と重なった状態を示している。

【0025】本発明では、同図に示すように内挿フレーム上で既内挿部分と現内挿部分が重なった場合に、次のようにして内挿画素値を定める。

【0026】(1) 重なった部分

内挿画素値 = (既内挿画素値 + 現内挿画素値) / 2

(2) 重ならない部分

内挿画素値 = 現内挿画素値

図3は内挿作業が終了した後に隙間が残った状態を示している。

【0027】この場合、本発明では、例えば図4に示すように、近隣画素値を用いて補間処理を行う。

【0028】図4において補間する画素1を●とすると、●の同列で一番近い既内挿画素は上の部分でX1、下の部分でX2である。同様に、同行で一番近い内挿画素値は、X3とX4である。そして、X1～X4と●との距離の逆数をR1～R4、画素Xの画素値をx(X)と現すと、本発明では、●の画素値を次のように補間する。
$$g(\bullet) = (g(X1) * R1 + g(X2) * R2 + g(X3) * R3 + g(X4) * R4) / (R1 + R2 + R3 + R4)$$
図5は本発明を実現する動画画像フレーム再生装置の構成を示すブロック図である。

【0029】ここでは、符号化方式としてH、261方式を想定し、画像をブロック分割しブロック毎に動きベクトルおよび動き元部分と現ブロックとの誤差を符号化しているものとする。また、1ブロックの大きさは、16画素×16画素で、水平方向にNブロック、垂直方向にMブロック、あるものとする。

【0030】同図に示すように、まずブロック誤差復号回路11がブロック毎にブロック情報からブロック誤差

を復号し、動きベクトル復号回路12がブロック毎にブロック情報から動きベクトルを復号する。

【0031】次に、動き元部分切り出し回路13が動きベクトル復号回路12により復号された動きベクトルに基づいて、フレームメモリ14より動き元部分を切り出し、加算器15においてこの切り出された動き元部分とブロック誤差復号回路11により復号されたブロック誤差とを加算し、現ブロックの画素値を再生する。

【0032】この後、フレーム内挿部16において、現ブロックによって内挿フレームの画素値を内挿する。

【0033】これは、まず復号された動きベクトルの長さを乗算器17により1/2にしたベクトルを用いて、はめ込み回路18がフレームメモリ19から内挿部分を切り出す。

【0034】次に、復号されたブロック誤差を乗算器20により1/2にしたものと動き元部分の画素値とを加算器21により加算し、この加算した値を用いて重なり処理回路22が画素の重なった部分を処理する。この処理方法については図2により説明した方法による。

【0035】このような重なり処理をした後、はめ込み回路18がフレームメモリ19の切り出された部分にはめ込む処理を行う。

【0036】このように合成されたフレームは、隙間処理回路23により隙間を補間する処理が行われる。この隙間の補間処理方法については図4および図5に示した方法による。

【0037】そして、全ての処理が終わった後、内挿フレームをディスプレイに表示し、次に再生フレームを表示する。再生フレームは、次の処理のため、フレームメモリ14に蓄えられているのでここから呼び出すことができる。

【0038】図6および図7はフレーム内挿部16における処理を示すフローチャートである。図6は主に重なり処理を、図7は主に隙間処理を示している。

【0039】まず、内挿作業を行う前に、フレームメモリ19を0クリアする(ステップ601)。

【0040】続いて、ブロックの動きベクトルの長さを1/2にしたベクトルと動き元部分の画素値に復号されたブロック誤差の1/2を加算した値とを順次読み込み(ステップ602、ステップ605～ステップ608)、フレームメモリ19から内挿部分を切り出し(ステップ603)、重なり処理を行いながらフレームメモリ19上に内挿値を書き込んでいく(ステップ604)。ここで、重なりがあるかどうかは、切り出した内挿部分の画素値が0かどうかで判断する。

【0041】全てのブロックの情報を読み込んだ後、フレームメモリ19中に値0の画素があるかどうかを調べ(ステップ701～ステップ709、ステップ711～ステップ714)、もしあればこれを隙間と判断し補間処理を行う(ステップ710)。

【0042】なお、図5に示すフレームメモリ19は、動きベクトルの検索範囲が全画面でなければ、1フレーム分である必要はない。例えば、検索範囲が±15画素であれば、メモリ容量は3ブロックライン分でよい。また、H、261方式のように符号化をGOB(画像の1/6、11ブロック×3ブロックラインのこと)単位で行う場合には、4ブロックライン(2GOB+1ブロックライン)分でよい。次に、符号化情報として動きベクトルを用いない符号化方式に本発明を適用した実施例を説明する。

【0043】図8はこの実施例に係る動画像フレーム再生装置の構成を示すブロック図である。

【0044】ここでは、2枚の連続して再生されたフレームがフレームメモリ24、25に蓄えられるものとし、その間にK枚のフレームを内挿するものとする。

【0045】まず、画像を一画素以上の大きさのブロックに分割し、最初のブロックをk=1とする。

【0046】そして、動きベクトル検索回路27がフレームメモリ25よりフレームのブロックを順次呼びだし、フレームメモリ24を用いて現ブロックの動き元部分を検索する。検索範囲は、例えば現ブロックの上下左右10画素とし、検索方法は、例えば現ブロックとの差分の2乗平均値をとって最小となる部分を選ぶものとする。

【0047】このようにして得られた現ブロックの動き元部分と動きベクトルを用いて内挿画像・ベクトル計算回路28が次のように内挿画像部分の画素値と内挿ベクトルを計算する。

【0048】

内挿画像部分=現ブロック×k/(K+1) + 動き元部分×(K+1-k)/(K+1)

内挿ベクトル=動きベクトル×(K+1-k)/(K+1)

次に、この内挿画像部分と内挿ベクトルを用いてフレーム内挿部29にてフレームを合成し、k枚目の内挿フレームとして表示する。なお、フレーム内挿部29は図5に示したフレーム内挿部16と同様の処理を行う。

【0049】次に、k=k+1とし、k=Kとなるまで以上の処理を繰り返す。

【0050】最後に、再生フレームをフレームメモリ25より呼び出して表示し、次の処理のためにフレームメモリ24に蓄える。

【0051】なお、動きベクトル検索回路27における動き元部分検索は、k=1のときの動きベクトルを記憶しておくならば、k=2からは必要ない。

【0052】また、内挿画像・ベクトル計算回路28における内挿画像部分と内挿ベクトルの計算は、上述したように線形である必要はなく、2枚の再生フレームの間を埋める動き軌跡は特に限定されない。

【0053】さらに、図5に示した実施例においても、内挿フレームの前後の再生フレームと復号した動きベク

トルを記憶するメモリがあれば、上述した方法を適用してK枚のフレームを合成することが可能である。外挿枚数Kは各時点において可変であってもよい。

【0054】次に、コマ落しされたフレームよりも前の再生フレームのみを用いてフレームを合成する実施例を説明する。

【0055】図9はこの実施例に係る動画像フレーム再生装置の構成を示すブロック図である。

【0056】ここでは、現再生フレームをもとに、K枚のフレームを外挿するものとする。同図に示す装置では、まずブロック誤差復号回路30がブロック情報から各フレームのフレーム誤差を復号する。また、動きベクトル復号回路31がブロック情報から各フレームの動きベクトルを復号し、これを動きベクトルメモリ32に蓄える。

【0057】次に、動き元部分切り出し回路33が動きベクトルを用いてフレームメモリ34から動き元部分を切り出す。そして、加算器35により動き元部分にブロック誤差を加算して、現再生フレームを表示するとともに、フレームメモリ34に蓄える。

【0058】このとき、同時に1枚目の外挿フレームを合成する。これは、まず動きベクトルを乗算器36により $-k/(K+1)$ 倍し、加算器35から出力される誤差を加えた動き元部分とともに、フレーム合成部37により外挿部分へはめ込みを行う。フレーム合成部37は図5に示したフレーム内挿部16と同様の処理を行う。

【0059】そして、全てのブロックについて再生および外挿作業を終えた後、現再生フレームに続いて1枚目の外挿フレームを表示する。

【0060】続いて、 $k=2$ とし、2枚目以降のフレームを外挿する。これは、まず動き元部分切り出し回路33がフレームメモリ34より順次ブロックを切り出し、これを内挿画像部分とする。ここで、フレームメモリ34には、すでにブロック誤差を加算した画素値が蓄えられているので、ここでは誤差加算は行わない。

【0061】そして、ベクトルメモリ32から現ブロックに対応した動きベクトルを呼び出し、動きベクトルを乗算器36により $-k/(K+1)$ 倍した内挿ベクトルと内挿画像部分を用いて、フレーム合成部37により外挿部分へはめ込みを行う。

【0062】全てのブロックについて外挿作業を終えた後、k枚目の外挿フレームを表示する。

【0063】そして、 $k=k+1$ とし、 $k=K$ となるまで以上の処理を繰り返す。

【0064】なお、本実施例では、外挿枚数Kは現時点において可変であってもよい。

【0065】また、 $K=1$ で固定の場合はフレームメモリ32は不要である。

【0066】さらに、本実施例は符号化情報に動きベクトルが含まれる場合について説明したが、図8に示した

方式を用いれば、その他の符号方式にも適用可能である。

【0067】次に、フレームを合成する際に内挿（前後のフレームを使って合成する。）と外挿（過去のフレームのみを使って合成する。）とを切り替えて用いる実施例について説明する。

【0068】図10はこの実施例に係る動画像フレーム再生装置の構成を示すブロック図である。

【0069】ここでは、簡単のため再生フレームの間に挿入する合成フレームは1枚とする。また、H、261方式を想定し、ブロックごとにイントラ／インターのフラグを符号化しているものとする。

【0070】同図に示すように、符号化されたフレームの再生においては、まず各ブロックのフラグをフラグ復号回路38により復号し、フラグがイントラの場合には、ブロック誤差復号回路39によりブロックの画素値を復号し、ベクトルメモリ40に“intra”であることを記録する。一方、フラグがインターの場合には、ブロック誤差復号回路39によりブロック誤差を、動きベクトル復号回路41により動きベクトルをそれぞれ復号し、ベクトルメモリ40に動きベクトルを蓄え、切り出し回路42が動きベクトルを用いてフレームメモリ43から動き元部分を切り出し、ブロック誤差を加算して画素値を再生する。

【0071】この際、内挿／外挿判定回路44が現ブロックによる合成を内挿とするか外挿とするかの判定を行う。この判定方法としては、例えば、イントラブロック→外挿、インターブロック→内挿とすればよい。

【0072】そして、外挿の場合には、切り出し回路45がフレームメモリ43から現ブロック部分を切り出し、ベクトルメモリ40からブロックに対応した動きベクトルを呼び出し $-1/2$ 倍し、これらを内挿画像部分と内挿ベクトルとして、フレーム合成部46で外挿部分へはめ込みを行う。フレーム合成部46は図5に示したフレーム内挿部16と同様の処理を行う。尚、ベクトルメモリ40より呼び出した値が“intra”であれば、現ブロックによる合成は行わず、隙間として残し、補間作業によって画素値を合成する。

【0073】一方、内挿の場合は図5に示した実施例と同じであるので説明を省略する。

【0074】そして、すべてのブロックについて再生および合成作業を終えた後、挿入する合成フレームを表示し、続いて現再生フレームを表示し、次の処理のためフレームメモリ43に蓄える。

【0075】なお、各ブロックにおける内挿／外挿の判定方法としては、イントラ／インターフラグを用いない場合には、次のような例が考えられる。

【0076】（1）ブロック誤差の絶対値和があるスレッシュホールドレベルを超えた場合は外挿。

【0077】（2）動きベクトルのブロック間差分の絶

対値があるスレッショールドレベルを超えた場合は外挿。

【0078】また、内挿／外挿の判定は、フレーム単位で行うこともできる。その場合は、次のような例が考えられる。

【0079】(1) シーンチェンジを検出した場合は外挿。

【0080】(2) 再生フレームの時間間隔が一定時間(例えば0.5秒)を越えた場合は次フレームを待たずに外挿によってフレーム挿入を行う。

【0081】最後の例は、フレーム内挿による時間遅延を避ける手段となる。

【0082】さらに、挿入枚数が複数枚の場合に、次フレームの到着を待つ間は、外挿によって初めのフレームを合成し、次フレームが到着した時点から後のフレーム合成に内挿を用いれば、画像表示時間の遅れを最小限に止めることができる。

【0083】次に、図11は図6に示したフレーム合成における隙間処理の他の例に示すフローチャートである。

【0084】同図に示す処理は、図5に示したフレーム内挿部16において重なり処理を行いながら、動き元部分の周辺の画素値を使って隙間の一部を同時に処理するものである。なお、ここでは、ブロックの大きさは16画素×16画素とする。

【0085】この処理では、まず内挿作業を行う前にフレームメモリFとはめ込み可否フラグメモリGを0クリアする(ステップ1101)。

【0086】続いて、ブロックの内挿ベクトルと、動き元部分に周辺aライン分を付け加えた(16+2a)画素×(16+2a)画素の大きさの復号値を順次読み込み(ステップ1102、ステップ1104～ステップ1107)、本来のはめ込み位置である16画素×16画素の部分は図6に示した処理と同様の処理を行い、はめ込み済みの画素についてはフラグメモリGを1とする。周辺部分については、はめ込みがされていない限り、補間値としてフレームメモリFに保存する。aは、例えば2とする(ステップ1103)。

【0087】このように、動き元部分の周辺の画素値を補間値に用いることで、はめ込みの際発生する隙間をほぼ埋めることができる。そして、すべてのブロックの情報を読み込んだ後、フレームメモリF中に値0の画素があるかどうかを調べ、もしあればこれを隙間と判断し、図7に示した補間処理を行う。

【0088】なお、図11によるフレーム合成方法は、動き元部分に周辺部を加えた画素部分を切り出す点を除

けば、図5に示したフレーム内挿部16と同様に各実施例に適用可能である。例えば、図8に示した実施例において、動きベクトル検索回路27により切り出される現ブロックと動き元部分を、周辺aライン分の画素を付け加えた部分とすれば、フレーム内挿部29に本実施例を適用することができる。

【0089】

【発明の効果】以上説明したように本発明の動画像フレーム再生方式によれば、動画像の符号化において発生するコマ落しを内挿する際に、動きがスムーズで視覚的に自然な内挿画像を得ることができる。

【図面の簡単な説明】

【図1】 本発明の原理を説明するための図である。

【図2】 本発明によるフレーム内挿方法において生じる画素の重なりを示した図である。

【図3】 本発明によるフレーム内挿方式において生じる隙間の補間方法の一例である。

【図4】 本発明によるフレーム内挿方法において生じる隙間の補間方法を示す図である。

【図5】 本発明の一実施例に係る動画像フレーム再生装置の構成を示すブロック図である。

【図6】 本発明に係る重なり処理の動作を示すフローチャートである。

【図7】 本発明に係る隙間補間処理の動作を示すフローチャートである。

【図8】 本発明の他の実施例に係る動画像フレーム再生装置の構成を示すブロック図である。

【図9】 本発明の他の実施例に係る動画像フレーム再生装置の構成を示す図である。

【図10】 本発明の他の実施例に係る動画像フレーム再生装置の構成を示すブロック図である。

【図11】 本発明に係る重なり処理の他の例を示すフローチャートである。

【図12】 従来の動画像フレーム再生装置の構成を示す図である。

【図13】 従来のフレーム内挿方法を説明するための図である。

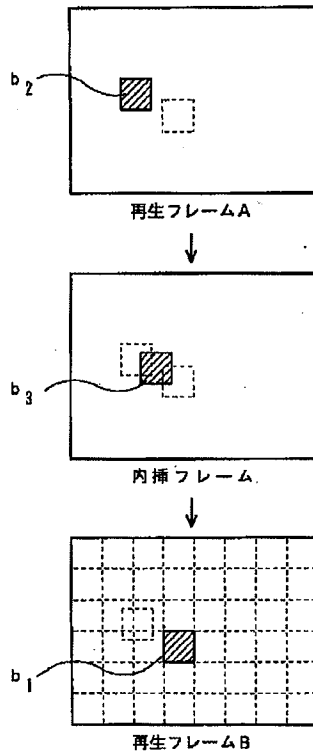
【図14】 従来のフレーム内挿方式を説明するための図である。

【符号の説明】

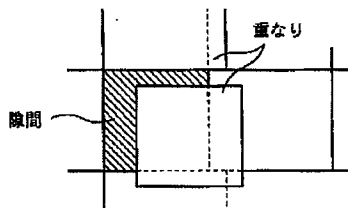
11…ブロック誤差復号回路、12…動きベクトル復号回路、13…動き元部分切り出し回路、14…フレームメモリ、15…加算器、16…フレーム内挿部、17…乗算器、18…はめ込み回路、19…フレームメモリ、20…乗算器、21…加算器、22…処理回路、23…隙間処理回路。



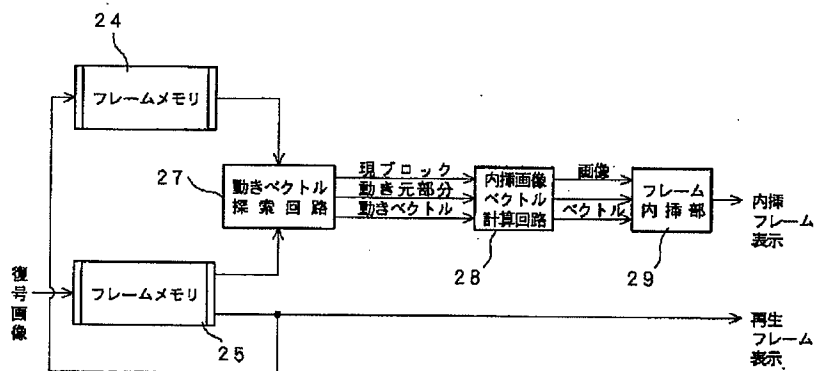
【図1】



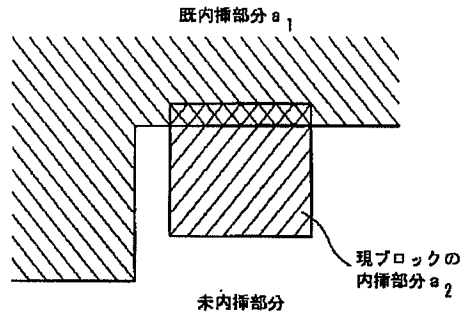
【図3】



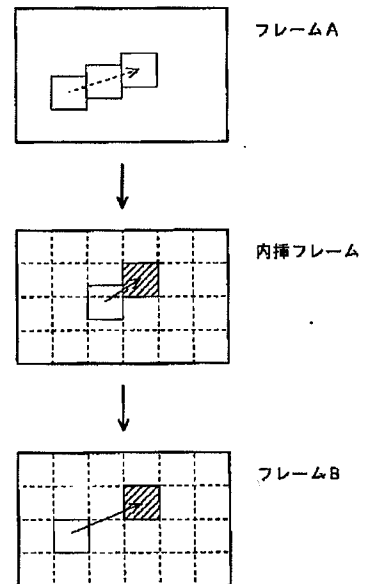
【図8】



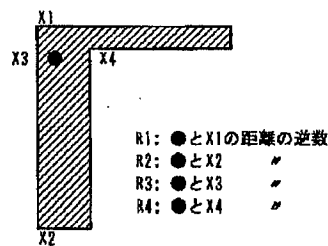
【図2】



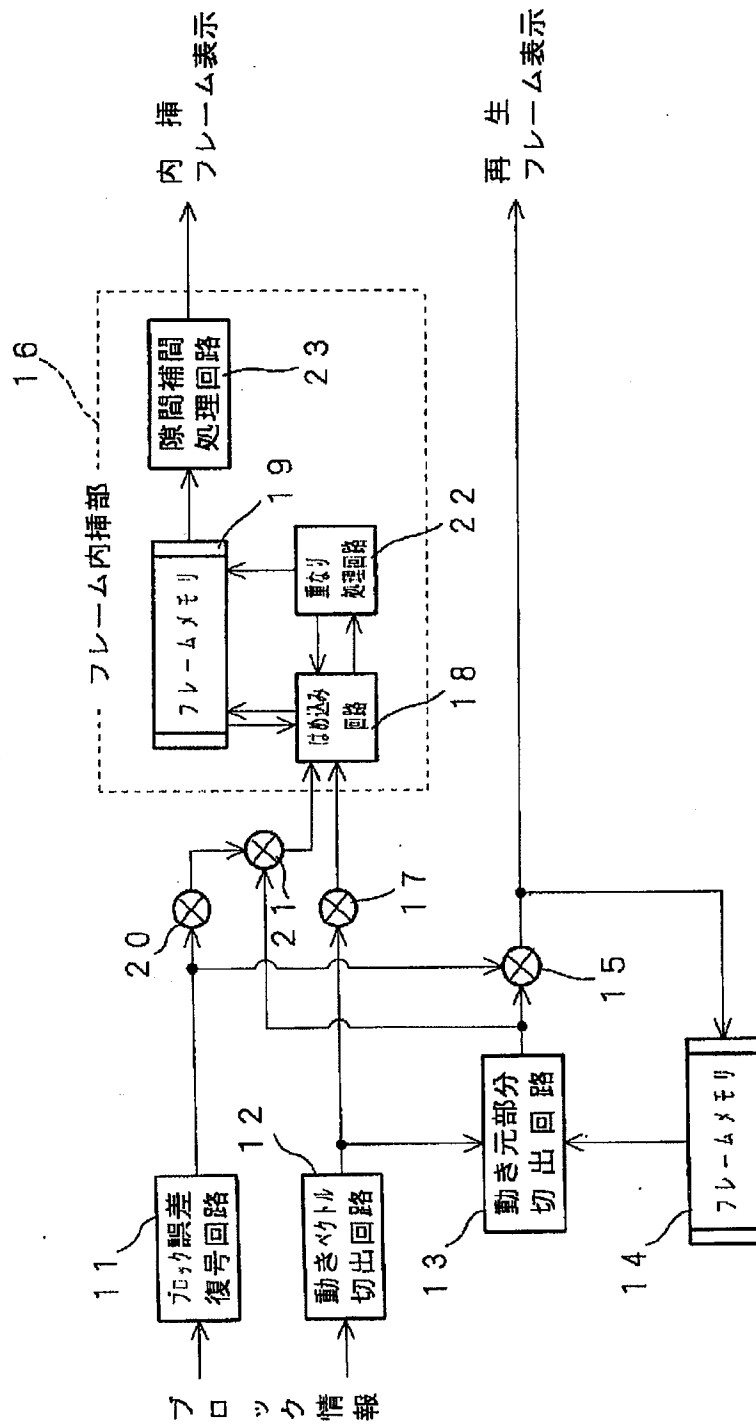
【図14】



【図4】

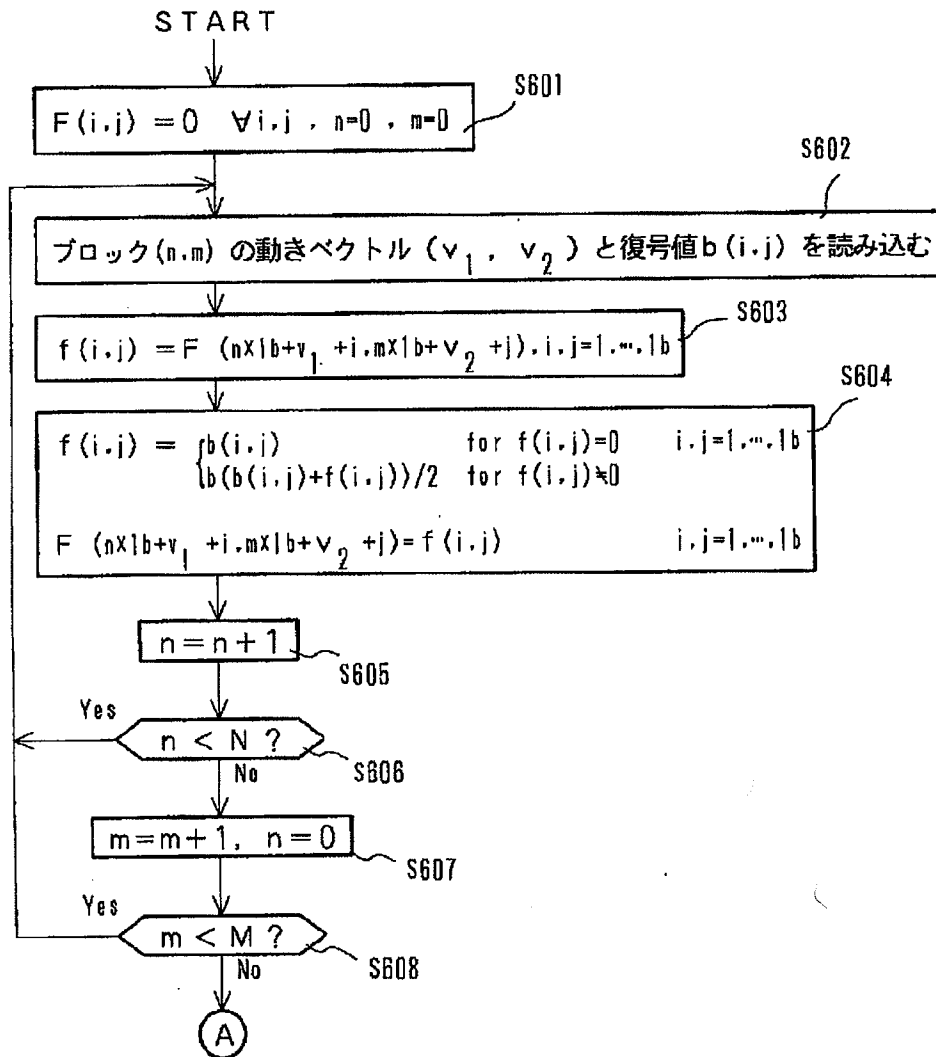


-8-

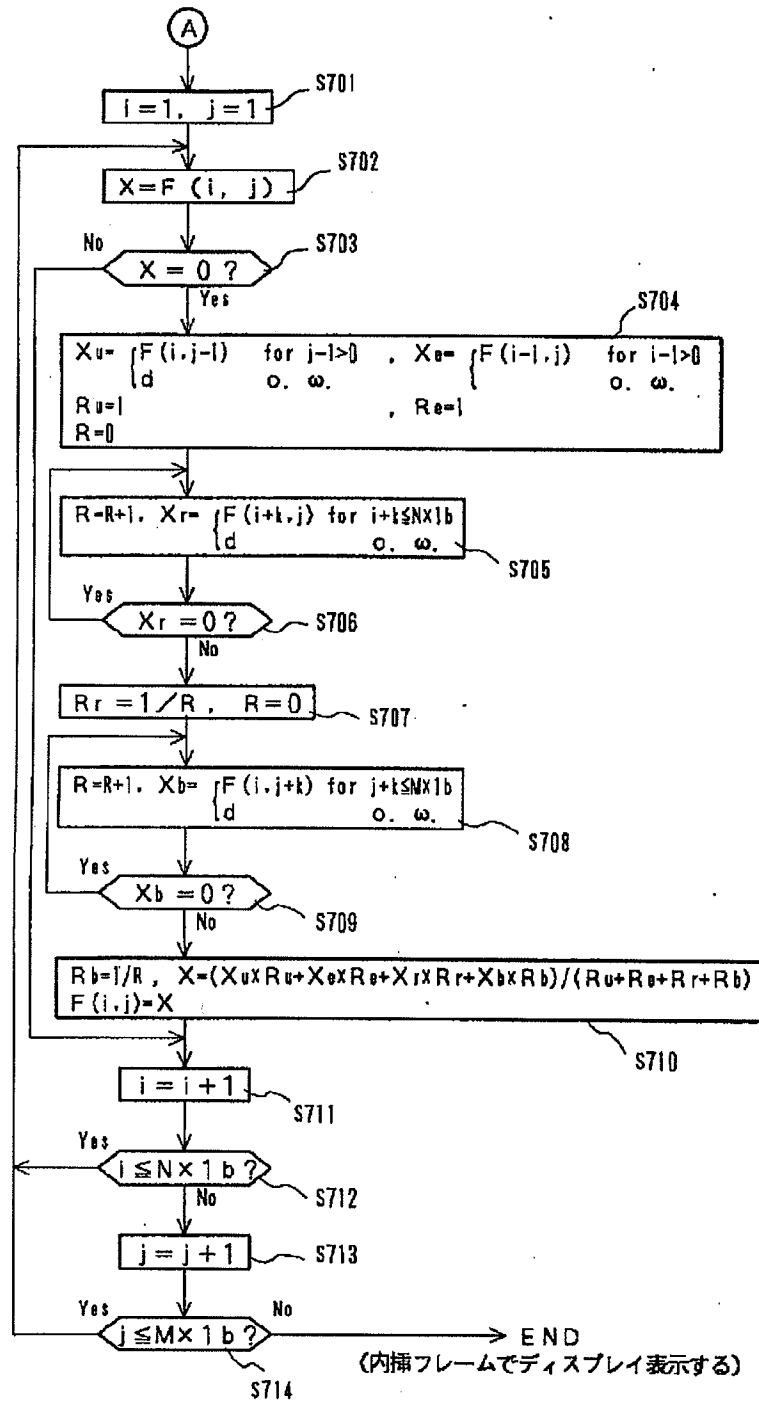


【図6】

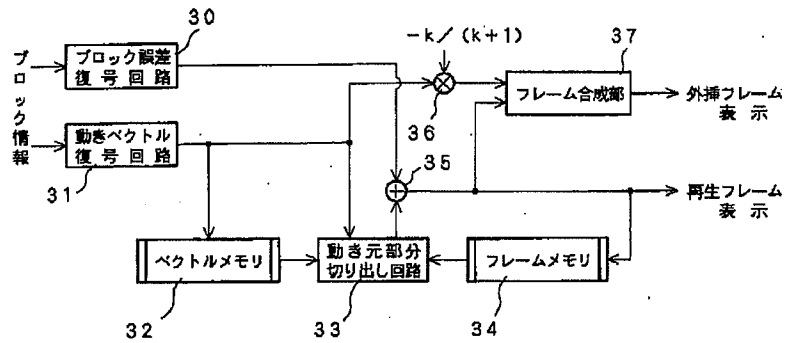
$N, M$ : 水平・垂直方向のブロック数  
 $F(i, j)$ : フレームメモリ  $j=1, \dots, NX|b, j=1, \dots, MX|b$   
 $(v_1, v_2)$ : 動きベクトル  
 $b(i, j)$ : 内挿に用いる復号値のブロック  $i, j=1, \dots, 1b$   
 $f(i, j)$ : フレームメモリから切り出したブロック  $i, j=1, \dots, 1b$   
 $n, m$ : ブロックアドレス  
 $X, Xu, Xe, Xb, Xr$ : 補間に用いる画素値  
 $R, Ru, Re, Rb, Rr$ : 補間に用いる画素間の距離の逆数  
 $d$ : デフォルト画素値



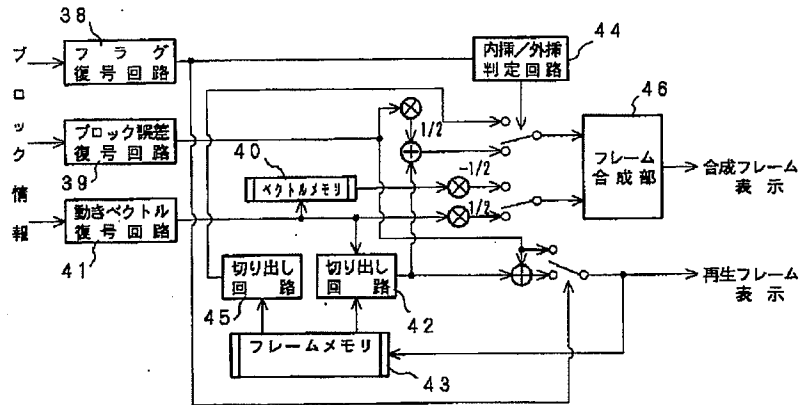
【図 7】



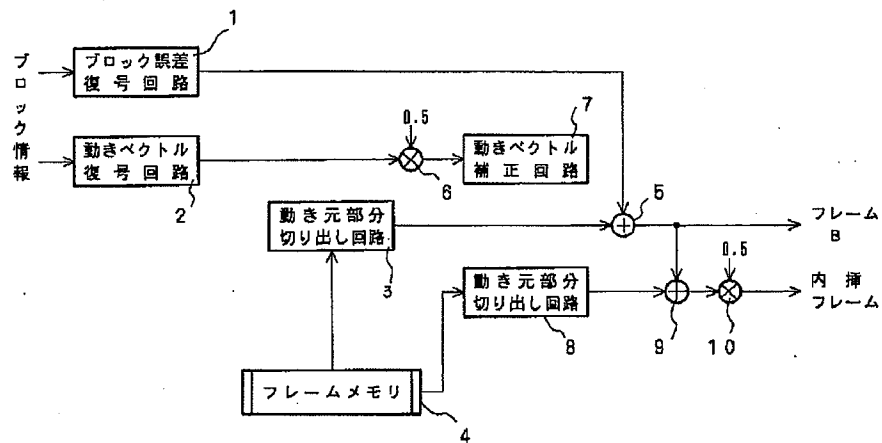
【図9】



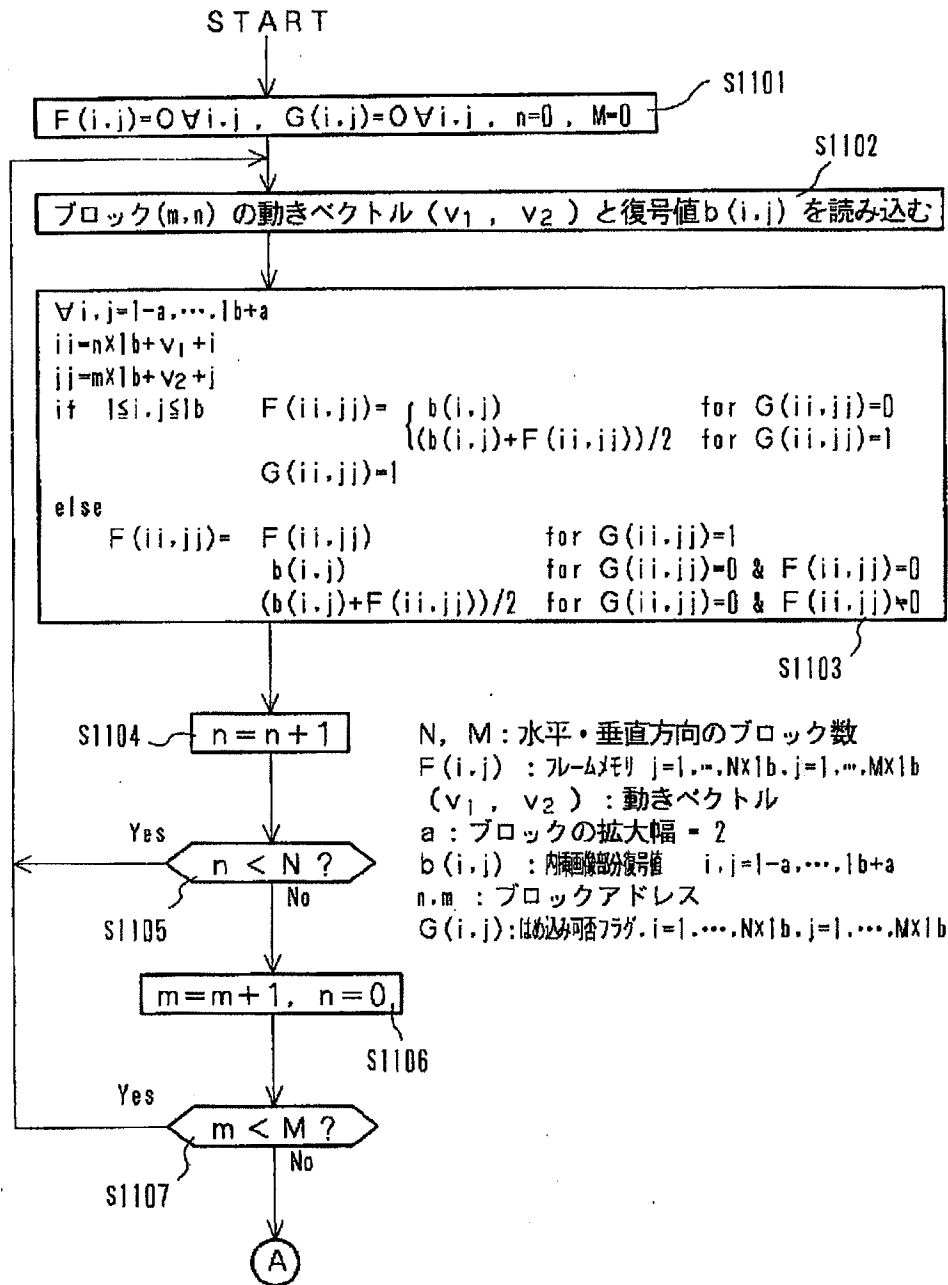
【図10】



【図12】



【図11】



【図13】

